IEEE HOME | SEARCH IEEE | SHOP | WEB ACCOUNT | CONTACT IEEE

Membership Publications/Services Standards Conferences Careers/Jobs

IEEE Xplore®
RELEASE 1.5

Welcome
United States Patent and Trademark O

Help FAQ Terms IEEE Peer Review Quick Links

Welcome to IEEE Xplore®

- ○ Home
- ○ What Can I Access?
- ○ Log-out

Tables of Contents

- ○ Journals & Magazines
- ○ Conference Proceedings
- ○ Standards

Search

- ○ By Author
- ○ Basic
- ○ Advanced

Member Services

- ○ Join IEEE
- ○ Establish IEEE Web Account
- ○ Access the IEEE Member Digital Library

Print Format

Your search matched **4** of **988420** documents.

A maximum of **4** results are displayed, **15** to a page, sorted by **Relevance** in **descending** order.
You may refine your search by editing the current search expression or entering a new one the text box.
Then click **Search Again**.

`((cepstral)and (frame*)) and(peak*)` Search Again

Results:
Journal or Magazine = **JNL** Conference = **CNF** Standard = **STD**

1 **An efficient and scalable 2D DCT-based feature coding scheme for remote speech recogniti**
Qifeng Zhu; Alwan, A.;
Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE Internatio Conference on , Volume: 1 , 7-11 May 2001
Page(s): 113 -116 vol.1

[Abstract] [PDF Full-Text (352 KB)] **IEEE CNF**

2 **On the use of variable frame rate analysis in speech recognition**
Qifeng Zhu; Alwan, A.;
Acoustics, Speech, and Signal Processing, 2000. ICASSP '00. Proceedings. 2000 IEEE Internationa Conference on , Volume: 3 , 5-9 June 2000
Page(s): 1783 -1786 vol.3

[Abstract] [PDF Full-Text (384 KB)] **IEEE CNF**

3 **A mixed-phase homomorphic vocoder**
Quatieri, T., Jr.;
Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '79. , Volum 1979
Page(s): 56 -59

[Abstract] [PDF Full-Text (82 KB)] **IEEE CNF**

4 **A novel pitch estimation technique using the Teager energy function**
Abu-Shikhah, N.; Deriche, M.;
Signal Processing and Its Applications, 1999. ISSPA '99. Proceedings of the Fifth International Sym on , Volume: 1 , 22-25 Aug. 1999
Page(s): 135 -138 vol.1

[Abstract] [PDF Full-Text (288 KB)] **IEEE CNF**

AN EFFICIENT AND SCALABLE 2D DCT-BASED FEATURE CODING SCHEME FOR REMOTE SPEECH RECOGNITION

Qifeng Zhu and Abeer Alwan

Department of Electrical Engineering, UCLA
Los Angeles, CA 90095
{qifeng,alwan}@icsl.ucla.edu

ABSTRACT

A 2D DCT-based approach to compressing acoustic features for remote speech recognition applications is presented. The coding scheme involves computing a 2D DCT on blocks of feature vectors followed by uniform scalar quantization, run-length and Huffman coding. Digit recognition experiments were conducted in which training was done with unquantized cepstral features from clean speech and testing used the same features after coding and decoding with 2D DCT and entropy coding and in various levels of acoustic noise. The coding scheme results in recognition performance comparable to that obtained with unquantized features at low bitrates. 2D DCT coding of MFCCs together with a method for variable frame rate analysis [Zhu and Alwan, 2000] and peak isolation [Strope and Alwan, 1997] maintains the noise robustness of these algorithms at low SNRs even at 624 bps. The low-complexity scheme is scalable resulting in graceful degradation in performance with decreasing bit rate.

1. INTRODUCTION

In certain applications, such as speech recognition over the World Wide Web and dictation via low-power cellular phones, there is a need for client-server recognition systems in which the recognition system is located at a remote server and the client performs less complex tasks such as feature extraction or signal compression.

There are two approaches to the remote recognition problem. The first involves coding the speech signal, transmitting the data, decoding the bitstream and performing feature extraction for ASR (e.g., [6]) or the bitstream is directly transformed to ASR feature vectors (e.g., [1]). In the second approach, which is the focus of this paper, feature extraction is first performed, then the features are compressed and transmitted to a remote server for recognition. This approach may be preferred if one had access to uncompressed speech signals and no playback is necessary, since transmitting the feature vectors can greatly reduce the bit rate with relatively low-computational cost.

In [3], the authors evaluated uniform and non-uniform scalar quantization, vector quantization, and product-code quantization of ASR features and achieved bit rates between 1.2 kbps-10.4 kbps with corresponding degradation in recognition performance. Ramaswamy and Gopalakrishnan [7] compressed acoustic features for speech recognition by using linear prediction and a two-stage vector quantizer to quantize prediction errors resulting in a 4 kbps scheme with nearly no loss in recognition performance. In [8], the authors used first order linear prediction and entropy constrained scalar quantization to compress Mel-Frequency Cepstral Coefficients (MFCCs), which are commonly used as a front end for ASR. The scalable, in bit rate, system resulted in good recognition accuracy at less than 1 kbps. None of these feature coding schemes, however, were evaluated in the presence of acoustic noise.

In this paper, a two-dimensional (2D) Discrete Cosine Transform (DCT) based coding method is used to compress ASR feature vectors. The 2D DCT is widely used in image compression and has been used to compress line spectral pairs (LSP) for speech coding [4]. We will show that the 2D DCT together with entropy coding can be used to compress MFCC feature vectors effectively at low bitrates.

2. OVERALL DESCRIPTION OF THE ALGORITHM

At the client, speech is first segmented into frames, features are computed for each frame, and then blocks of features are generated. A 2D DCT is then performed on each block and components with the lowest energy are set to zero. This is followed by scalar quantization, run-length and Huffman encoding. A block diagram of the encoder is shown in Figure 1. At the receiver, decoding and IDCT are performed and feature vectors corresponding to each frame are inputted to the ASR system. Only the feature vectors are encoded and sent to the recognition server; the first and second derivatives are computed at the server based on the recovered features. In the following sections, each of these operations is explained.

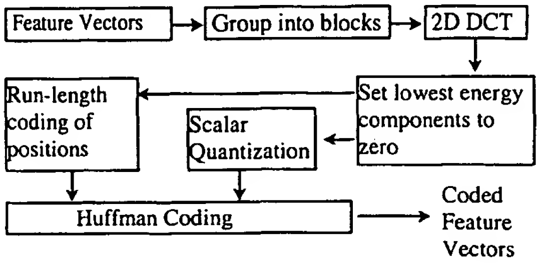


Figure 1. Block diagram of the DCT and entropy encoder.

0-7803-7041-4/01/$10.00 ©2001 IEEE

2.1 2D DCT of Feature Vectors

Front-end processing for speech recognition systems converts the speech waveform into a sequence of feature vectors computed for 20-30 ms overlapping segments. A common set of feature vectors used for ASR are the MFCCs [2] which are computed by integrating an initial power spectrum estimate that is weighted by bandpass-filters whose bandwidths approximate those of auditory filters (typically 26 filters). A logarithmic function is then used to compress the magnitude of the power estimates, and the spectral estimate for each frame can be roughly decorrelated using a 1D DCT. The first component (c(0)), which is related to signal energy, is usually not considered for ASR but the following 12 DCT components with their first and second derivatives are.

Several techniques for making the MFCCs noise robust have been proposed such as liftering [5], and peak isolation (enhancement of the peak-to-valley ratio) [9]. In addition, in [10] we showed that variable frame rate (VFR) processing can decrease the average frame rate of transmitting feature vectors while improving recognition performance in noise. VFR is based on using energy weighted distance metrics.

In this paper, feature vectors are transmitted in a stream of blocks and for each block a 2D-DCT is applied. The motivation for performing a 2D-DCT is to exploit inter-frame correlations among feature vectors which are attributed to underlying temporal redundancies in the speech signal. Signals are first windowed with 25 ms overlapping Hamming windows (window shift is 10 ms). A block of features is generated by stacking together feature vectors for 12 frames. Hence, each block is 12×12 where the columns are MFCC vectors for each frame and the rows are MFCCs of the same order in 12 frames. If we denote each NxN block of feature vectors as a matrix, U, then the 2D DCT transformed matrix, V, can be computed as: $V=AUA^T$. The elements of the matrix A are:

$$a(i,j) = \begin{cases} \frac{1}{\sqrt{N}}, & i = 0, 0 \le j \le N\text{-}1 \\ \sqrt{\frac{2}{N}}\cos\frac{\pi\,(2j+1)i}{2N}, & 1 \le i \le N\text{-}1, 0 \le j \le N\text{-}1 \end{cases}$$

(Eq. 1)

The DCT results in energy compaction, with energy concentrated at the low-order components, which makes effective compression possible. Since MFCCs are generated by computing a DCT in the first place, intra-frame correlation of MFCCs is small, even after truncation and liftering. Inter-frame correlation of MFCCs of the same order, however, is high, so after the 2D-DCT, energy is compacted to the lower order components (in the row direction) resulting in large values for the first column in each block.

Figures 2 and 3 illustrate the energy compaction property of the DCT. Figure 2 shows the MFCCs for the digit /one/ as spoken by a female talker. The result of computing the 2D DCT on three 12x12 blocks of that utterance is shown in Figure 3. Note that the beginning of each block (corresponding to its first column) has the highest energy components.

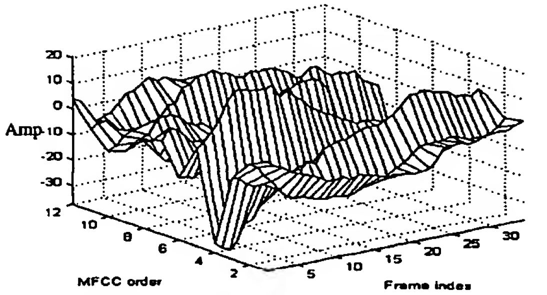


Figure 2. MFCCs for the digit /one/ spoken by a female talker.

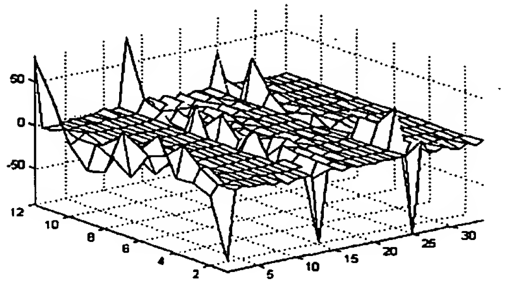


Figure 3. Three 12 × 12 blocks of MFCCs after 2D DCT for the same digit shown in Figure 2.

To reduce the bit rate, the lowest energy elements in each block are set to zero. We define α as the ratio of the 2D DCT size (144) to the number of components in each block which are not set to zero. We illustrate the distortion effects from this operation with an example. Consider the MFCC transformed blocks in Figure 3. If we set to zero the lowest energy components such that α= 8, and then perform a 2D IDCT, we obtain the MFCC features shown in Figure 4. Note that the general shape of the MFCCs is preserved but finer details are not.

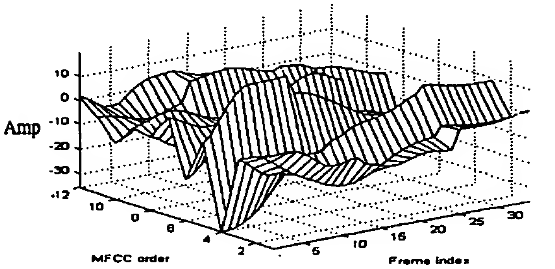


Figure 4. MFCCs for the same digit used in Figure 2 when α=8 (after performing a 2D IDCT).

The nonzero components in each block are then quantized using uniform scalar quantization.

2.2 Quantization, Run-Length and Huffman Coding of DCT Blocks

Since the first column of each block has the highest energy components, we assign β-1 bits for quantizing the components of all columns except the first one; the first column is assigned β bits per component.

Run length encoding of the position of non-zero components is performed. Each 2D DCT block is converted into a one-dimensional signal by concatenating the columns in each block. For the first 12 numbers (which constitute the first column of the 2D-DCT block) we label the zero components and for the rest, we label the non-zero components. This is because the components of the first column of each block are often not set to zero. Consequently, the number of labeled components is often smaller than the number of non-zero components. The run lengths are computed between pairs of two adjacent labeled components. The run length of the first labeled component is computed as its absolute position minus one. One extra run length of -1 is used as the symbol for the end-of-block (EOB). Based on the statistics of the run length and MFCC quantized values, one can compute the number of bits per point needed for Huffman coding. The overall bit rate of the coding scheme can be approximated as follows:

$$\mathit{block\_rate}*[\mathit{bits\_per\_point}*\mathit{number\_of\_nonzero\_points} + \mathit{bits\_per\_run\_length}*(\mathit{number\_of\_labeled\_points}+1)] \qquad \text{(Eq. 2)}$$

The coding scheme is scalable allowing the user to choose the appropriate bit rate. In the event of heavy internet traffic, for example, the number of bits assigned to each component (β) could be decreased and/or the DCT preservation rate (α) increased, resulting in lower overall bit rates.

3. RECOGNITION RESULTS

Digit recognition experiments were done with HTK2.1 with the speaker-dependent TI46 database (8 males and 8 females, 12.5 kHz sampling rate). For each digit a 4-state left-to-right Hidden Markov Model (HMM) with 2 Gaussian mixtures was trained using 160 utterances. Two steps of Maximum Likelihood (ML) and Expectation Maximization (EM) training, and diagonal covariance matrices were used. Silence portions were not included. Several feature vectors were compared in terms of recognition performance: 1) MFCCs, 2) Peak isolated MFCCs (MFCCP) [9], 3) Variable frame rate MFCCs with peak isolation (VFR_MFCCP) [10], and 4) DCT coded-decoded version of MFCCs, MFCCPs, and VFR_MFCCPs.

The Hamming window length was 25ms, and window shift was 10ms. Training and testing were done with MFCCs (or modified MFCCs) together with their first and second derivatives. Training was performed on un-quantized feature vectors from clean speech while testing was done with signals corrupted by various levels of additive speech-shaped noise. Testing was performed with features after being coded and decoded using the techniques described above with 480 balanced utterances for the same talkers (the first 3 utterances from each talker from the test database). The models were re-trained with peak isolation and VFR in Experiments 2-4.

Recognition results are shown in Tables 1-4 as a function of SNR. In Table 1, results are shown for unquantized MFCCs, MFCCPs, and VFR_MFCCPs. The best performance is obtained with VFR_MFCCPs. The ratio of the average frame rate of VFR_MFCCP to a fixed rame rate version is 1:1.7 [10].

For the 2D DCT and entropy coding scheme, we experimented with several values of α (2,4,6,8) and β (4,5,6,7,8). At β=4 bits/component, recognition results were significantly worse than testing with unquantized MFCCs. At β=8 bits/component, recognition results were as good as using β=7 bits/component. When β=5, 6, and 7 bits/component, the recognition accuracy/bitrate tradeoff was best for α=6, 4, and 2, respectively. These results are shown in Tables 2-4 for the three feature vectors (MFCC, MFCCP, and VFR_MFCCP). The corresponding bit rates are 1248, 2057 and 3783 bps (Table 2), 1087, 1770, and 3291 bps (Table 3), and 624, 1030, and 1936 bps (Table 4). Note that even though some of feature components were set to zero and the remaining components were represented by few bits, the scheme maintained recognition accuracy which is comparable to that obtained with unquantized features. Also note that the degradation in recognition accuracy is less when using peak-isolated MFCCs than with MFCCs. The reason for this is that inter-frame correlations are higher for the MFCCPs, thus resulting in a higher degree of energy compaction in the 2D DCT blocks. This is most striking for VFR_MFCCP where even at a bitrate of 624 bps, recognition results are significantly improved over the baseline system with unquantized MFCCs at low SNRs. Table 5 shows an example of bit distribution for three different (α,β) pairs for the TI46 database using variable frame rate analysis with MFCCP. The overall bitrate is computed using Eq. 2 with average values obtained from the test database. The block rate is 4.9 blocks/second.

Table 6 illustrates the graceful degradation in recognition performance with decreasing bitrate. The table shows recognition results as a function of SNR when β=6 bits/component and α varies between 2 and 8.

We also evaluated the scheme using the TIDIGIT speaker-independent database (80 talkers for training and 32 different talkers for testing) and similar trends were observed. For that database, a scheme with α=4 and β=6 results in recognition performance which is comparable to that with unquanitzed MFCCP and VFR_ MFCCP features as shown in Table 7.

SNR:	20dB	15dB	10dB	5dB	0dB
MFCC	99.2	98.1	92.5	66.3	34.0
MFCCP	98.5	97.3	92.3	75.2	44.0
VFR_MFCCP	99.2	98.8	97.3	88.3	61.0

Table 1. Digit recognition accuracy (in percent) as a function of SNR for unquantized feature vectors (MFCC, MFCC with peak isolation, and MFCC with variable frame rate and peak isolation) for the TI46 database.

SNR:	20dB	15dB	10dB	5dB	0dB
α=6, β=5	98.1	98.1	80.8	52.7	30.4
α=4, β=6	99.6	97.9	89.2	62.7	31.3
α=2, β=7	99.4	98.5	92.1	65.8	32.9

Table 2. Recognition accuracy when using MFCCs after coding/decoding by the 2D DCT and entropy coding scheme at 3 bitrates: 1248, 2057 and 3783 bps for rows 1-3, respectively.

SNR:	20dB	15dB	10dB	5dB	0dB
α=6, β=5	99.2	97.5	91.0	74.6	44.0
α=4, β=6	98.8	97.1	91.5	74.2	43.1
α=2, β=7	98.5	97.3	92.1	74.6	44.0

Table 3. Recognition accuracy when using MFCCPs after coding/decoding by the 2D DCT and entropy coding scheme at 3 bitrates: 1087, 1770, and 3291 bps for rows 1-3, respectively.

SNR:	20dB	15dB	10dB	5dB	0dB
α=6, β=5	97.9	97.7	93.1	83.1	55.4
α=4, β=6	98.8	98.3	96.7	86.7	56.3
α=2, β=7	99.2	99.0	97.3	87.9	59.4

Table 4. Recognition accuracy when using VFR_MFCCPs after coding/decoding by the 2D DCT and entropy coding scheme at 3 bitrates : 624, 1030, and 1936 bps for rows 1-3, respectively.

(α, β)	average number of labeled points per block	bits per run-length (avg.)	number of nonzero points per block	bits per point (avg.)	overall bit rate (bps)
6, 5	16.86	3.48	24	2.72	624
4, 6	27.49	3.17	36	3.33	1030
2, 7	62.13	2.23	72	3.53	1936

Table 5. An example of the distribution of bits for three different (α,β) pairs after Huffman coding using VFR_MFCCPs with the TI46 database.

SNR:	20dB	15dB	10dB	5dB	0dB
α=2	99.2	98.3	96.9	88.5	59.0
α=4	98.8	98.3	96.7	86.7	56.3
α=6	98.8	97.7	93.5	82.5	51.0
α=8	98.1	97.3	92.5	77.3	46.9

Table 6. Graceful degradation in recognition accuracy as the bitrate decreases (α=2-8). VFR_MFCCPs after coding/decoding by the 2D DCT and entropy scheme are used. β=6.

SNR:	20dB	15dB	10dB	5dB	0dB
MFCCP	98.1	94.7	87.1	80.9	65.2
MFCCP+DCT	98.4	94.7	86.2	81.5	65.2
VFRMFCCP	99.4	98.1	93.7	89.0	75.9
VFRMFCCP +DCT	99.4	98.2	93.4	89.0	76.2

Table 7. Digit recognition accuracy using the TIDIGIT database at different SNRs with α=4 and β=6 for MFCCP, and VFR_MFCCP with and without 2D DCT and entropy coding/decoding.

4. SUMMARY AND CONCLUSION

In this paper, a 2D DCT-based approach is used for coding feature vectors to achieve a scalable scheme with graceful degradation in recognition performance at the lower rates. The low-complexity scheme maintains the robustness of unquantized features in noise. When using MFCCs together with a method for variable frame rate analysis [10] and spectral peak isolation [9], error rates are significantly lower than those with unquantized MFCCs even at 624 bps for an isolated digit recognition task. While the method was tested for MFCC-based features, it could be applied to other ASR feature vectors as well. It could also be easily extended to continuous ASR tasks which is the focus of our current work.

The current version of the technique introduces a block-sized delay (approximately 120 ms for the fixed frame rate version and 204 ms for VFR). For the Internet, or any packet switching network, this delay may not be critical since packetization delay is inevitable. In fact, if a packet, for example, contains at least one block of coded features then the coding delay will not be noticeable. Delay could be reduced by using smaller block sizes. Future work will examine the effects of block size on recognition performance.

Acknowledgment

Work supported in part by NSF, HRL and Broadcom through the UC MICRO program, and ST Microelectronics.

5. REFERENCES

[1] S. Choi; H. Kim; H. Lee and R. Gray "Speech recognition method using quantized LSP parameters in CELP-type coders". Electron. Lett. , Vol. 34, no.2, IEE, 1998, p.156-7.

[2] S. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," IEEE Trans. on ASSP, Vol. 28, No. 4, 1980, p. 357-366.

[3] V. Digalakis, L. Neumeyer, and Perakakis. M "Quantization of Cepstral parameters for Speech Recognition over the World Wide Web," IEEE JSAC, Vol. 17. No. 1 Jan. 1999, p. 82-90.

[4] Farvardin, N and Laroia, R. "Efficient encoding of speech LSP parameters using the discrete cosine transformation". Proc. ICASSP 1989, Vol. 1, p 168-171.

[5] B. Juang, L. Rabiner, and J. Wilpon, "On the use of bandpass liftering in speech recognition," IEEE Trans. ASSP, Vol. 35, pp. 947-954, July 1987.

[6] B. Lilly and K. Paliwal, "Effect of speech coders on speech recognition performance," Proc. ICSLP 1996, Vol.4, p.2344-47.

[7]. G. Ramaswamy, and P. Gopalakrishnan, "Compression of Acoustic Features for Speech Recognition in Network Environments," Proc. IEEE ICASSP 1998, p. 977-980.

[8] N. Srinivasamurthy, A. Ortega, Q. Zhu, and A. Alwan, "Towards Efficient and Scalable Speech Compression Schemes for Robust Speech Recognition Applications," Proc. IEEE ICME 2000, p. 249-52 Vol.1.

[9] B. Strope and A. Alwan, "A model of dynamic auditory perception and its application to robust word recognition." IEEE Trans. on SAP, Vol. 5, No. 2, p. 451-464, Sep. 1997.

[10] Q. Zhu and A. Alwan, "On the use of variable frame rate analysis in speech recognition," Proc. IEEE ICASSP 2000, Vol. III, p. 1783-1786.

ON THE USE OF VARIABLE FRAME RATE ANALYSIS IN SPEECH RECOGNITION

Qifeng Zhu and Abeer Alwan

Department of Electrical Engineering, UCLA
Los Angeles 90095, USA
{qifeng, alwan}@icsl.ucla.edu

ABSTRACT

Changes in spectral characteristics are important cues for discriminating and identifying speech sounds. These changes can occur over very short time intervals. Computing frames every 10 ms, as commonly done in recognition systems, is not sufficient to capture such dynamic changes. In this paper, we propose a Variable Frame Rate (VFR) algorithm. The algorithm results in an increased number of frames for rapidly-changing segments with relatively high energy and less frames for steady-state segments. The current implementation used an average data rate which is less than 100 frames per second. For an isolated word recognition task, and using an HMM-based speech recognition system, the proposed technique results in significant improvements in recognition accuracy especially at low signal-to-noise ratios. The technique was evaluated with MFCC vectors and MFCC vectors with enhanced peak isolation [4].

1. INTRODUCTION

In most speech-processing systems, speech signals are first windowed into frames; frames are typically 20-30 ms in duration and the frame step size is 10 ms. This is especially true for HMM-based automatic speech recognition (ASR) systems. The justification for such a segmentation is that speech signals are non-stationary and exhibit quasi-stationary behavior at the shorter durations.

It is known, however, that certain acoustic attributes of the speech signal can be manifested at very short durations (see for example, [3]). Such attributes may be critical for the identification and discrimination of speech sounds.

In this study, we propose a variable frame-rate (VFR) approach for analyzing speech signals. The technique results in an increased number of frames when the spectral characteristics of the signal change significantly and less frames otherwise. The frame step size can be as low as 2.5 ms. The algorithm can be implemented such that the average data rate of the system is the same, less, or greater than the fixed data rate approach that is typically used in ASR systems.

For an isolated word, HMM-based, recognition task, the proposed technique results in significant improvements in recognition accuracy especially at low signal-to-noise ratios. The technique was evaluated with MFCC vectors and MFCC vectors with enhanced peak isolation [4].

The technique proposed in this paper differs from the VFR techniques in [2] and [5]. Both papers did not evaluate their systems in the presence of background noise. In addition, in [2], the focus was on using VFR to reduce the data rate of the system while keeping the frame step size at 10 ms, and thresholds were chosen in an ad hoc manner. In [5], a theoretically-motivated VFR system was proposed, but the evaluation was only done with a DTW recognition system and did not show improvement in recognition accuracy.

2. MOTIVATION AND PRELIMINARY EXPERIMENTS

In a previous study [1], we examined the acoustics and perception of the place-of-articulation for the highly confusable nasal consonants /m, n/ in pre-stressed syllable-initial position with the vowels /a, i, u/. The database was collected at UCLA and consists of speech tokens by 2 male and 2 female talkers with 8 repetitions per syllable (192 tokens in total). The sampling rate was 16 kHz. Perceptual experiments were conducted both in quiet, and in the presence of additive white Gaussian noise (AWGN) and speech-shaped noise. Results showed that formant transitions, in general, play a larger role in identifying place than the murmur. Specifically, perceptual thresholds were correlated with the duration and relative amplitudes of the formant, especially F2, transitions which in turn were vowel dependent. For example, the duration of the F2 transitions in /na/ syllables were the longest, as shown in Table 1, and with relatively high energy leading to very robust perception of the sound in noise. In addition, /ma/ syllables were robust even though F2 transition was short, but the amplitude of F2 relative to F1 was the largest of all syllables.

ma	mi	mu	na	ni	nu
19	20.8	16.6	57.5	19.3	12.9

Table 1. Average F2 transition in millisecond for different syllables. Measurements were done manually.

To compare human and machine nasal recognition, an experiment was conducted using the HMM-based ASR system from Entropics Inc. (HTK 2.0). Endpoint detection using energy and zero-crossing measures was used. Each HMM model had 6 states. Training was done with half of the utterances, and testing, with the other half. The feature vector used was the Mel-Frequency Cepstral Coefficients (MFCC) with first and second derivatives. The window (Hamming) length was 25 ms and the frame step size was 10 ms. The noise in all the experiments is additive speech shaped noise. If the system is trained and tested with clean data, high recognition accuracy is achieved (90 percent). If the system is trained with clean data and tested with noisy data, recognition scores deteriorate. For example, at a SNR of 3 dB, the recognition accuracy is 52 percent; the corresponding confusion matrix is shown in Table 2. The /Ca/ syllables are the most robust in noise and we attribute that result to the more pronounced formant transitions for those syllables. We speculate that the deterioration in recognition accuracy for /Ci/ syllables, in particular, is attributed to their very short and weak formant transitions.

	ma	mi	mu	na	ni	noo	Correct rate (%)
ma	13	0	0	3	0	0	81.2
mi	0	0	0	2	6	8	0.0
nu	1	0	2	8	0	5	12.5
na	0	0	0	16	0	0	100
ni	0	1	0	1	8	6	50.0
nu	0	0	0	5	0	11	68.8

Table 2. Nasal recognition results. Trained with clean data, and tested with additive speech shaped noise at a SNR=3dB.

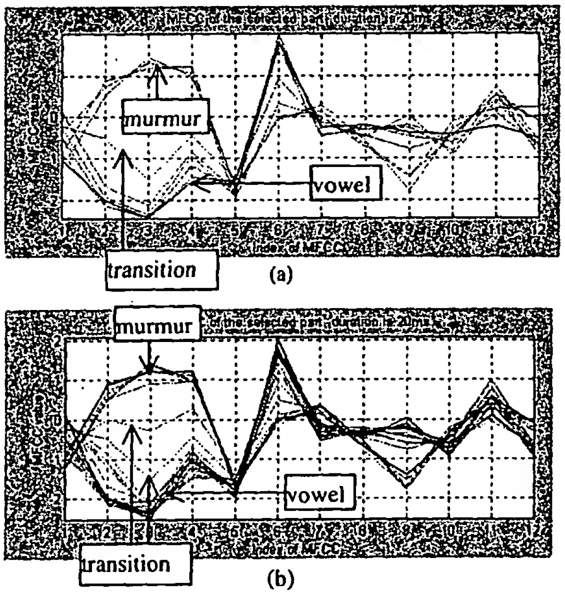


Figure 1. MFCC vectors around the transition of a /ma/ utterance. (a) Window step size = 10ms. (b) Window step size = 2.5ms.

A frame step size of 10 ms may not be sufficient to capture dynamic changes. To illustrate this point, Figure 1 shows plots of MFCC vectors along a 100 ms segment surrounding the formant transition region in a /ma/ syllable. The frame length is 20 ms, but the frame step size is 10 ms in (a) and 2.5 ms in (b). Speech is pre-emphasized and MFCC vectors are liftered. Note that the murmur and steady-state region of the vowel are represented by (perhaps an unnecessarily large) number of MFCC vectors, while the critical formant transition region (13 ms) is only represented by one vector with a 10 ms frame step size and 2 (distinct) vectors when the step size is reduced to 2.5 ms.

3. VARIABLE FRAME RATE (VFR) METHOD

3.1 The Algorithm

From the analysis described above, it is clear that computing frames every 10 ms is not adequate for representing rapidly changing segments although it is sufficient for representing relatively steady and long ones.

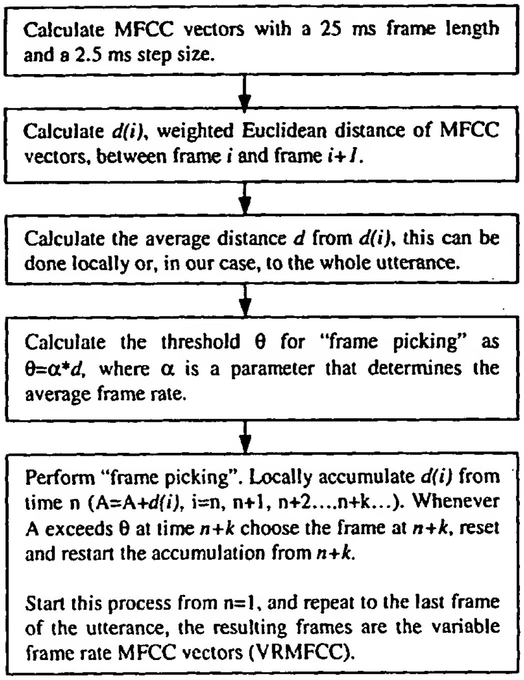
One solution to this problem is increasing the frame rate, but this would unnecessarily increase the computational load of ASR systems and is not needed for steady segments. Instead, we propose a variable frame rate method in which the frame rate varies as a function of the spectral characteristics of the signal.

Using MFCC feature vectors, the variable frame rate algorithm is implemented as shown in Figure 2.

First, speech is analyzed with frame lengths of 25 ms (Hamming window) and a step size of 2.5 ms. We refer to these frames as the "dense frames". Second, the difference (*d(i)*, where *i* is the time index,) between every two adjacent "dense frames" is calculated. The average of these differences is then calculated over the whole utterance. Third, based on the weighted differences, some frames are kept and others are discarded. In particular, "dense frames" around a formant transition will be kept, while at the steady part of the signal, frames will be picked sparsely.

It is important to note that the distance *d(i)* is calculated as the *energy weighted* Euclidean MFCC distance: first the Euclidean distance of the MFCC vectors of two adjacent frames are calculated, then it is weighted by (E - β), where E is the log energy of that frame, and β is a constant offset. This is different from the method proposed in [2] where the Euclidean MFCC distance was used. Energy weighting is important so that segments which exhibit changes but are low in energy are discarded, since they may not be noise robust. Our previous experiments have shown a clear relationship between the energy of formant transitions and perceptual noise robustness. In addition, our pilot ASR experiments using Euclidean MFCC distance did not yield high recognition accuracy in noise.

The two parameters α, the threshold, and β, log energy offset, are chosen experimentally. The choice of α will determine the average data rate. For example, if α is 4 (ratio of the 10 ms step size and the dense step size of 2.5 ms), then the resulting total number of frames will be nearly the same as that in a front-end with a frame step size of 10 ms. If α is larger than 4, then the average data rate will be less than 100 frames per second and vice versa. In our implementation, α was chosen to be 6.8. The log energy offset β was set to be the average E (over the entire utterance) divided by 1.5.



Calculate MFCC vectors with a 25 ms frame length and a 2.5 ms step size.

Calculate *d(i)*, weighted Euclidean distance of MFCC vectors, between frame *i* and frame *i+1*.

Calculate the average distance *d* from *d(i)*, this can be done locally or, in our case, to the whole utterance.

Calculate the threshold θ for "frame picking" as θ=α**d*, where α is a parameter that determines the average frame rate.

Perform "frame picking". Locally accumulate *d(i)* from time n (A=A+*d(i)*, i=n, n+1, n+2....n+k...). Whenever A exceeds θ at time *n+k* choose the frame at *n+k*, reset and restart the accumulation from *n+k*.

Start this process from n=1, and repeat to the last frame of the utterance, the resulting frames are the variable frame rate MFCC vectors (VRMFCC).

Figure 2. Flow chart of computing variable frame rate MFCC vectors.

3.2 An Example of VFR Analysis

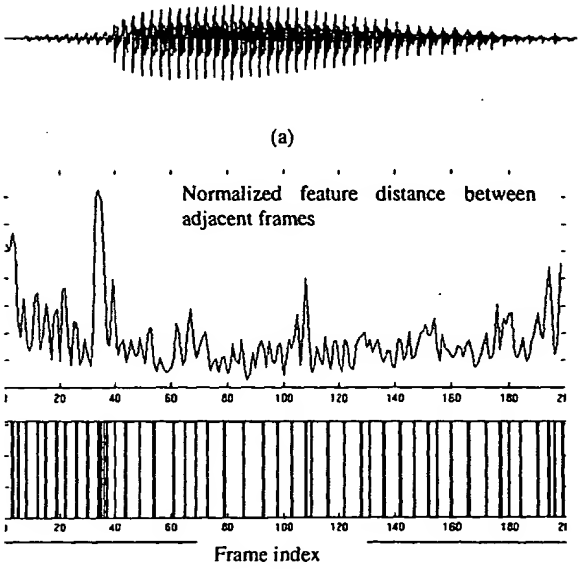
Figure 3 illustrates how frames are picked for the utterance /ma/ as spoken by a male speaker. Part (a) shows a time waveform of the utterance. The upper part of (b) plots *d(i)*, the weighted feature distance between two adjacent frames - with a step size of 2.5 ms - and the lower part shows the result of the frame-picking algorithm where each bar indicates that a frame has been chosen for recognition. Note that near the transition region from the consonant to the vowel *d(i)* is large. For this example, 50 out of 200 dense frames are picked. Around the transition region, all the dense frames (spaced by 2.5 ms) are kept while in the steady-state part of the vowel, only 3-4 frames, out of 20 frames, are selected corresponding to a step size which is larger than 10ms.

3.3 Recognition with the VFR Front End

The variable frame rate method was used in ASR experiments using the nasal database described in Section 2, and the TIDIGITS database. In the experiments, the performance of the recognition system with two feature vectors were compared: MFCC, and MFCC vectors with peak enhancement [4] (hereafter referred to as MFCCP). First and second derivatives of these features were used. Training was done using clean data while testing was done with either clean or noisy data.

Results for the nasal recognition experiment are shown in Table 3. Clearly, the variable frame rate approach together with a method for enhancing spectral peaks, gives the best performance at low SNRs.

The VFR method was also used with the database "Studio Quality Speaker Independent Connected Digit Corpus" (TIDIGITS). Each left to right digit HMM model had 4 states, 2 mixtures, and a diagonal covariance matrix. 80 utterances from 80 speakers, (40 male and 40 female) were used to train each model. Test data were from the other 32 speakers (half male and half female).



(a)

(b)

Figure 3. (a) The waveform of a /ma/ utterance. (b) The upper panel is the normalized *d(i)* distance, and the lower panel shows which frames were kept.

	Clean	SNR=15dB	5dB	0dB
MFCC	90	89	68	34
MFCCP	96	91	77	68
VRMFCCP	100	96	81	71

Table 3. Percent correct rates from different front-ends using the nasal database.

We compared MFCC and MFCCP with their variable frame rate versions. The results are shown in Figures 4 and 5 and is summarized in Table 4. The results clearly illustrate that applying the VFR method to each feature vector improves recognition performance especially at low SNRs. Increasing time resolution for rapidly changing segments, while keeping the time resolution low for steady parts, results in improved robustness.

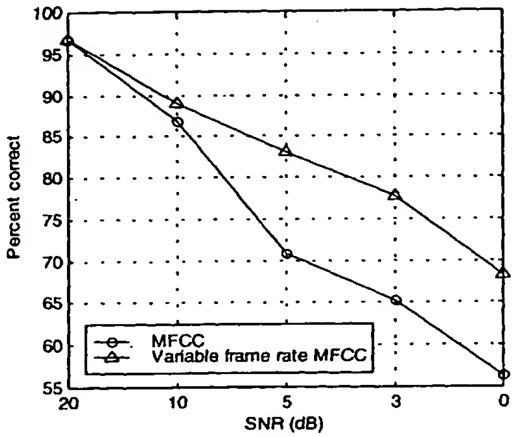


Figure 4. Recognition results expressed by word percent correct for MFCC and variable frame rate MFCC using the TIDIGITS database.

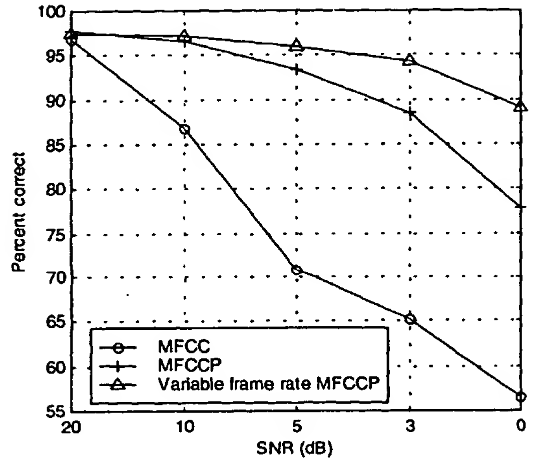


Figure 5. Recognition results expressed by word percent correct for MFCC, MFCC with peak isolation (MFCCP), and its variant frame rate version (VRMFCCP) using the TIDIGITS database.

Percent correct	SNR=20 dB	10dB	5dB	3dB	0dB
MFCC	96.87	86.83	70.85	65.20	56.43
MFCCP	97..81	96.55	93.42	88.40	77.74
VRMFCCP	97.49	97.18	95.92	94.36	89.03

Table 4. Recognition results summary for MFCC, MFCCP and VRMFCCP front ends using the TIDIGITS database.

4. SUMMARY AND CONCLUSION

Changes in spectral characteristics are important cues for discriminating and identifying speech sounds. These changes can occur over very short time intervals. Computing frames every 10 ms, as commonly done in ASR systems, is not sufficient to capture such dynamic changes. In this paper, we propose a Variable Frame Rate (VFR) algorithm. The algorithm results in an increased number of frames for rapidly-changing segments with relatively high energy and less frames for steady-state segments. It can be implemented such that the average data rate is the same, less, or more than a fixed 100 frames per second data rate. The current implementation used an average data rate which is less than 100 frames per second. For an isolated word recognition task, and using an HMM speech recognition system, the proposed technique results in significant improvements in recognition accuracy especially at low signal-to-noise ratios. The technique was evaluated with MFCC vectors and MFCC vectors with enhanced peak isolation [4].

The novel properties of the proposed VFR algorithm are 1) using energy weighted MFCC distance, 2) allowing a frame step size as low as 2.5 ms, and 3) a novel method for frame selection.

Acknowledgement

Work is supported in part by NSF and NIH.

5. REFERENCES

[1] Abeer Alwan, Jeff Lo, and Qifeng Zhu "Human and Machine Recognition of Nasal Consonants in Noise", Proceedings of the 14th International Congress of Phonetic Sciences, Vol. 1, p. 167-170, 1999.

[2] Ponting, K.M. and Peeling, S.M. "The use of variable frame rate analysis in speech recognition." Computer Speech and Language Comput. Speech Lang. (UK), vol.5, (no.2), April 1991. p.169-79.

[3] Kenneth Stevens, *Acoustic Phonetics*, MIT Press, 1998.

[4] Strope, B. and Alwan, A. 1997. "A model of dynamic auditory perception and its application to robust word recognition", IEEE Trans. on Speech and Audio Processing, Vol. 5, No. 5, p. 451-464.

[5] Young, S.J. and Rainton, D. "Optimal frame rate analysis for speech recognition." IEE Colloquium on Techniques for Speech Processing (Digest No.181), London, UK. 17 Dec. 1990, p.5/1-3.